Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 12 (20): 2156-2163, 2012
ISSN 1812-5654 / DOI: 10.3923/jas.2012.2156.2163

Fast Detection of Stealth and Slow Scanning Worms in Transmission Control Protocol

[1,2]Mohammad M. Rasheed, [1]Osman Ghazali and [1]Rahmat Budiarto
[1]School of Computing, College of Arts and Sciences, Universiti Utara Malaysia,
06010 UUM Sintok, Kedah, Malaysia
[2]Telecommunication Research Center, Information Technology Directorate,
Ministry of Science and Technology, Iraq

Abstract: Anti-virus systems and most current intrusion-detection systems are signature based technology. The problem in signature-based technology is that they can only detect a known worm with identified signatures that have been produced recently. The detection system must therefore be able to handle known and likewise, unknown threats but the false alarm is high false alarms when used anomaly detection system to detect unknown worms. This study developed a new technique that depended on the anomaly detection system to detect the stealth scanning worm by two sub techniques. The first sub technique is considered new failure connection messages that generated by stealth scanning worm and second sub technique is included multi threshold by considered the speed of worm spread for generated the threshold. The result of this study showed the proposed technique capable of detecting the stealth and slow scanning of Internet worm and faster than other methods without any false-positive warning, besides reduced the false-negative warning.

Key words: Internet worm detection, slow worm scanning, stealth scanning

INTRODUCTION

Currently, worms are widely regarded as a serious security threat. The Internet scanning worms spread in an automated way, which infected many host in the Internet in a very short time. 'Code-Red' worm incidents on July 19th 2001 that infected 36,000 hosts of hosts within fourteen hours (Paul, 2001). Anti-virus systems and most current intrusion-detection systems are signature based technology (Min and Gupta, 2009; Mohammed *et al*., 2010; Moskovitch *et al*., 2009; Zolkipli and Jantan, 2010), the problem in signature-based technology is that they can only detect a known worm with identified signatures that have been produced recently (Tang and Chen, 2005). Besides anti-virus, the firewalls can be used to detect worm signature and block the known worm packets (Muda *et al*., 2011; Yu *et al*., 2009), but this reactive response happens only after the worm already spread. The detection system must therefore be able to handle known and likewise, unknown threats (Nasir *et al*., 2008), but the false alarm is high false alarms when used to detect unknown worms (Meenakshi and Srivatsa, 2007). In addition, the rate of false alarms could be large and take long time to detect the worm. Where the false-negative alarm allows the worm to escape containment, while false positives may cause network outages by blocking normal traffic (Costa, 2006).

The worms used TCP to find the victim. TCP has six control flags in the TCP protocol. Each bit of a control flag gives to acknowledge to the other machine sides. The Fin Flag (FIN) sender transmits a FIN flag when it has no more data to transmit. The Synchronize flag (SYN) is used to synchronize the sequence number. The Reset Flag (RST) sends a packet with an RST flag when it wants to fail the connection. When the sender requests the receiver to deliver the data to the application program immediately, it puts a Push Flag (PSH). Acknowledgment Flag (ACK) means the TCP header includes the acknowledged sequence number. Normally, all packets except for the first packet in a connection have ACK flags. Urgent Flag (URG) means the packet includes some urgent data (Fukushima and Goto, 1999).

A TCP connection is always starting with the 3-way handshake, which establishes and negotiates the actual connection over which data will be sent. The whole session is begun with a SYN packet, then a SYN/ACK packet and finally, an ACK packet to acknowledge the whole session establishment (Jiang and Zhu, 2009).

Some worms used TCP to find the victim. In TCP worm scanning, there are two important conditions to transfer the worm from the infector machine to the victim. The first condition when the worm IP target address is used in a victim B. The second important condition is to transfer the worm from computer A to computer B when the port for computer B is open as shown in Fig. 1.

Corresponding Author: Mohammad M. Rasheed, School of Computing, College of Arts and Sciences, Universiti Utara Malaysia, 06010 UUM Sintok, Kedah, Malaysia

After that, computer A replicates itself to computer B as shown in Fig. 2 and closes the connection. When a TCP connection is closed, computer A sends FIN and computer B replies by ACK.

When the IP address is unused in the destination IP address; the router returned an ICMP Destination Unreachable to source IP (infector computer) (Ellis *et al*., 2004) (Fig. 3).

When the worm sends a SYN packet from the source IP address to a destination IP that is being used, but if the destination port is closed, then it returns the RST/ACK packet (Ellis *et al*., 2004) (Fig. 4).

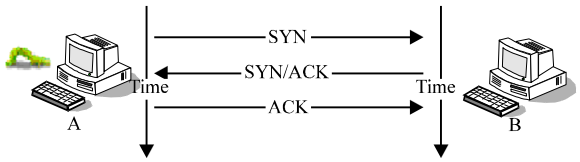

Fig. 1: TCP open connection

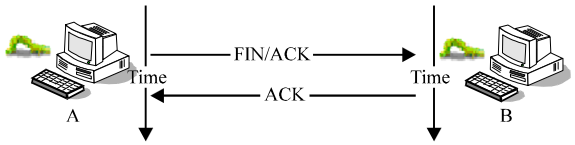

Fig. 2: TCP close connection

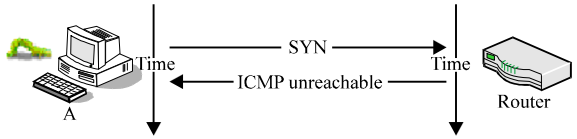

Fig. 3: SYN request status when the destination IP is unused

Whenever, a destination host does not reply, the router discards a packet due to a time-out, it will generate a Time Exceeded Type 11 ICMP (Dubendorfer *et al*., 2005), as shown in Fig. 5.

Also, there are worms use stealth attacks like Ramen worm that uses FIN scan (Jiang and Zhu, 2009). There are three types of stealthy scan in TCP protocol namely (FIN) scan, (FIN, URG, PSH) scan and (Null) scan. The null scan means that no flag is sent (De Vivo *et al*., 1999). In the study, they are called 'stealth' scans because they send a single flag to a TCP port without any TCP handshaking or additional packet transfers. This is a scan type that sends a single flag with the expectation of a single reply. In this FIN scan, TCP port is closed so the remote station sends an RST/ACK frame response to the FIN packet (Messer, 2007). The worms can use stealth scan to attack other machines (Hiestand, 2005). Figure 6 shows the

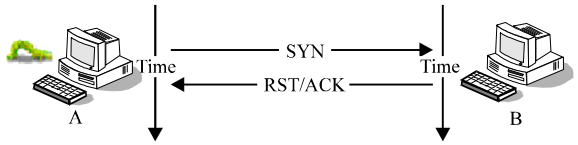

Fig. 4: SYN Request Status When Destination Port is Closed

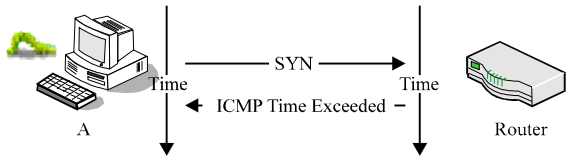

Fig. 5: Router reply for SYN when destination IP is not responded

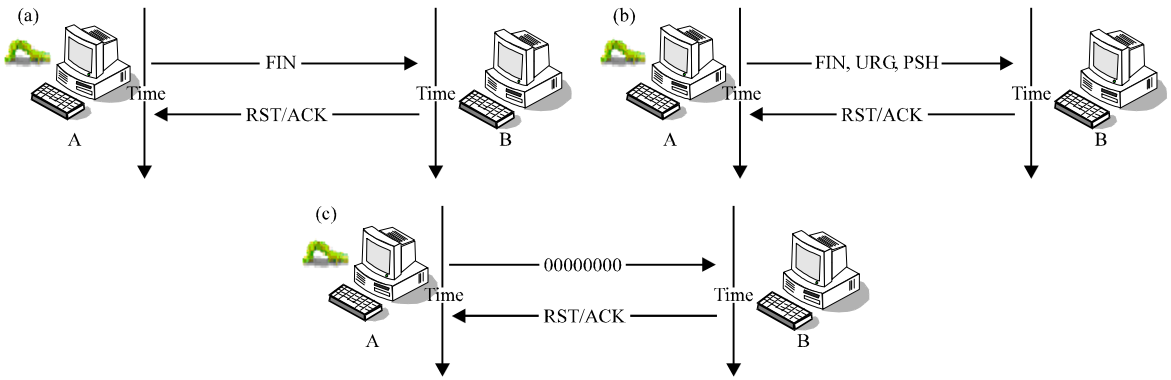

Fig. 6: TCP/Stealth (a) Fin, (b) FIN, URG, PSH and (c) 00000000 scanning when the port victim is closed

Table 1: Mechanisms analysis

Worm detection technology (message error)	Worm detection	Slow or stealthy worm detection	Speed	References
ICMP unreachable	-	-	Slow	Zou *et al.* (2003)
ICMP unreachable	-	-	Slow	Berk *et al.* (2003)
ICMP unreachable and RST	(√)	(√) but some worm cannot detect it fast	Fast	Yang *et al.* (2006)
ICMP unreachable and RST	(√)	(√)	Slow	Chen and Tang (2007)
ICMP unreachable, ICMP time Exceeded and RST	(√)	(√)	Faster than Yang algorithm	Proposed technique

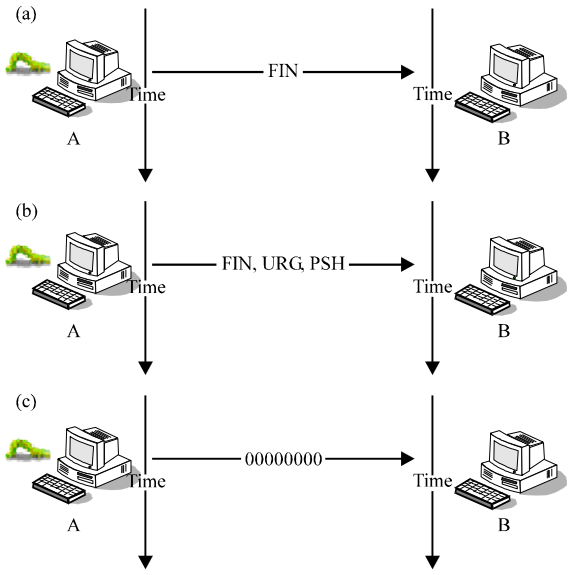

Fig. 7: TCP/Stealth Scanning (a) No respond for FIN scanning, (b) No respond for FIN, URG, PSH scanning and (c) No respond for 00000000 scanning When the Port of Victim is Opened

stealth scan sends request but the port is closed so the remote station sends an RST/ACK frame response.

SYN scan considers no response to indicate a filtered port, while a stealth scan treats the same as open or filtered (Lyon, 2009), as shown in Fig. 7.

RELATED WORK

Zou *et al.* (2003) proposed the design of a worm monitoring system. The monitoring system purposes to provide comprehensive monitoring data on a worm's activities for the early detection of the worm. They focused just on the ICMP message. Berk *et al.* (2003) proposed a monitoring system by collecting ICMP. They used a potentially unlimited number of collectors and analyzers. Yang *et al.* (2006) proposed algorithm that has two sub algorithms. The first that is 'short term algorithm' that work well to detect fast scanning worm. While the second, which is 'longer term algorithm' that detects stealthy scanning worm. The detection worm depended on ICMP Unreachable and RST/TCP. Chen and Tang (2007) analyzed the essential character of TCP-based worm's propagation that sending out a large number of TCP connection requests. They proposed an effective approach to detect network worms based on the number of failure connection received by the network routers. The approach can be divided into two phases: short term and longer term. This strategy may be work well on detecting uniform scanning worm and 'stealthy' worm. However, the impact on normal network activities has not been considered, as shown in Table 1 that includes the different mechanism analysis and the proposed technique.

DESIGN OF PROPOSED TECHNIQUE

The study uses the name of stealth scans of the Internet worm that sends a single frame to a TCP port without any TCP handshaking or additional packet transfers. Stealth scan sends a single frame with the anticipation of a single response (Yaqub, 2006). The failure connections that are received via TCP stealth worm scanning are ICMP Unreachable, ICMP Time Exceeded and RST/ACK messages. See Use Case analysis for stealth scanning worm for respond to the stealth scanning in Fig. 8.

There are three types of failure connections. The first failure connection is received when the Internet worm sends a request and the port is closed for destination. The infected machine receives RST/ACK. Design of Proposed Technique (DPT) increases the Counter of Failure Connection (CFC) when received a failure connection from destination IP address. The second failure connection is received when IP address is unused in the destination; the infected machine received ICMP Unreachable. The third failure connection is received when destination IP is filtered. The infected machine receives ICMP Time Exceeded. Once detecting the first failed connection packets, the time starts and DPT extracts the destination address from the packet, after that, creates the record in History of Connection (HC). The condition is important to reduce the false alarm; DPT increases or decreases CFC when receiving failure or success (SYN/ACK) connection from destination IP address that is not recorded in HC. DPT ignores the

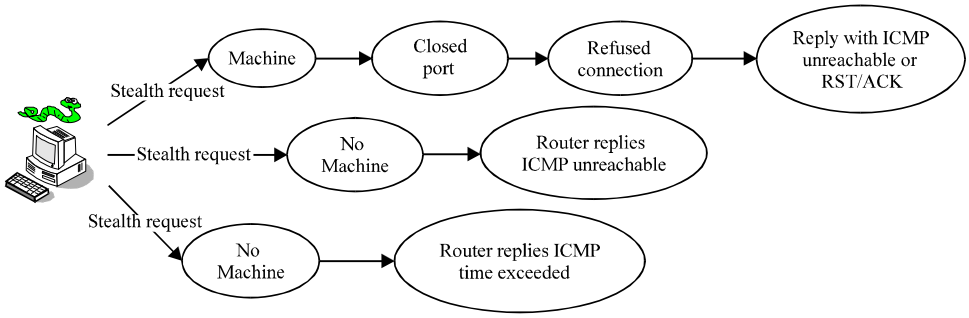

Fig. 8: Use case diagram for stealth scanning worm

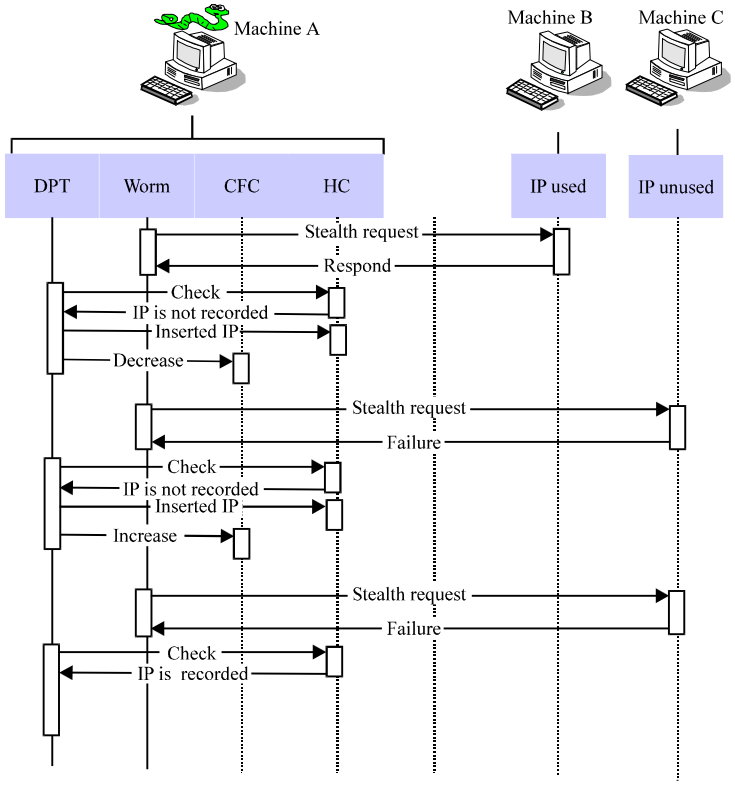

Fig. 9: Sequence diagram for stealth scanning worm

packet when the destination IP is recorded in the HC because the Internet scanning worm attack strategy is 'attacking different IP addresses'. See sequence analysis for stealth scanning worm in Fig. 9.

DPT removes the CFC every three days. DPT uses three days to detect slow scanning worm that generates low failure connections. DPT is capable of detecting the worm scanning that generates an average of failure connection, which is (0.7 and up)/min.

DPT has a multi threshold to detect the worm. When the failure is high, DPT detects the worm very fast, but when the worm has a low failure connection. In this case, DPT needs more time to detect it. Moreover, when the worm uses slow scanning in the UDP or TCP protocol, the result for the slow scanning is a low failure of connection. DPT can detect this type of worm scanning, because DPT has multi threshold and depends on the failures' message for UDP or TCP protocol. DPT starts with Threshold (T) equal β where (β = 101 failure connections/per one minute). After one minute, DPT calculates the Average of Failure Connection (AFC) to find the T.

$$AFC = CFC/Summation\ of\ the\ Time \quad (1)$$

where, summation of the time means the process time from first failure connection in CFC until last failure in CFC. The time depends on the minutes.

Where:

$$T = 2^{\wedge} (6.655 + 0.0495 (\beta - AFC)) \quad (2)$$

When the study compares DPT with Yang *et al.* (2006), the Yang's algorithm detects the Internet scanning worm if the failure connection is greater than 100/min failure connections by using short term algorithm. When the failure connections are greater than 3000/day failure connections, the Yang's algorithm detects this type of stealthy Internet scanning worm by using long term algorithm. DPT uses the same Yang's algorithm warning but DPT has multi threshold, depending on the average of failure connection. DPT depends on Yang *et al.* (2006) for calculating warning.

Scenario 1: When using Yang's algorithm to detect the Internet scanning worm that has 3001/day failure connections, Yang's algorithm detects the worm after one day. DPT can detect the worm also after one day:

- AFC = 3001/1440 (one day = 1440 min)
- AFC = 2.084027/one min

Then T will be:

$$T = 2^{\wedge} (6.655 + 0.0495 (\beta\text{-}AFC))$$

- $T = 2^{\wedge} (6.655+0.0495 (101\text{-}AFC))$
- $T = 2^{\wedge} (6.655+0.0495 (101\text{-}2.084027))$
- $T \approx 3001$ failure connections/day, DPT detects the worm by one day

Yang algorithm cannot detect the average of failure connection that is less than 2.08 failures, but DPT can detect the average of failure connection that is up to 0.7.

Scenario 2: When using Yang's algorithm to detect the Internet scanning worm that generates 101/minute failure connections, Yang's algorithm needs one minute-time process for detecting this worm. DPT can also detect it in one minute:

- AFC = 101/1
- AFC = 101/one minute

Then T will be:

$$T = 2^{\wedge} (6.655 + 0.0495 (\beta - AFC))$$

- $T = 2^{\wedge} (6.655 + 0.0495 (101 - AFC))$
- $T = 2^{\wedge} (6.655 + 0.0495 (101 - 101))$
- $T \approx 101$ DPT will detect the worm in one minute

DPT and Yang's algorithm detect the worm in one minute. DPT uses different threshold values over different time periods; therefore, DPT is faster than Yang's algorithm when the worm is less than 3001/day and up to 101/min failure connections. Moreover, DPT detects the worm when it has been less than 3001/day failure connections, unlike Yang's algorithm. DPT threshold depends on the average of failure connection to compute it. Where CFC is calculating the failure progress and T is the threshold to detect the worm. The condition is CFC equals or greater than T. If true, it detects the worm. Unlike Yang's algorithm, DPT is more dynamic in detecting the worm because it calculates the threshold every minute. Whenever the counter value does not exceed the threshold, DPT reads the next packet. See DPT in Fig. 10.

EVALUATION OF DPT AND YANG'S TECHNIQUE

The study evaluated DPT with Yang *et al.* (2006). The study showed DPT was faster than Yang *et al.* (2006) algorithm in all different of average failure connection of slow worms, because DPT had multi threshold, as shown in Table 2. Moreover, DPT depended on three failure messages and they are ICMP unreachable, ICMP Time Exceeded and RST/TCP messages, but Yang *et al.* (2006) algorithm depended on two failure messages and they are ICMP unreachable and RST/TCP messages.

The result was DPT faster than Yang *et al.* (2006) by two procedures, the first DPT had multi threshold and the second depended on three of failure message connections for calculated the threshold. The faster detection means reduced the false-negative alarm.

Validation of DPT: The machine was uninfected by any malware and the machine was installed with DPT. The user used the Internet in uninfected machine for browsing different websites and chats such as YouTube, Facebook, Yahoo Messenger and others during the time for validation. The machine operating system used was Microsoft 2000 professional Service Pack 4. Moreover, the machine was connected with a network device by Celcom that supports the Internet by mobile wireless and the broadband speed was 3.6 MB sec^{-1}. DPT was examined on an uninfected machine for ten days to validate the false-positive alarm.

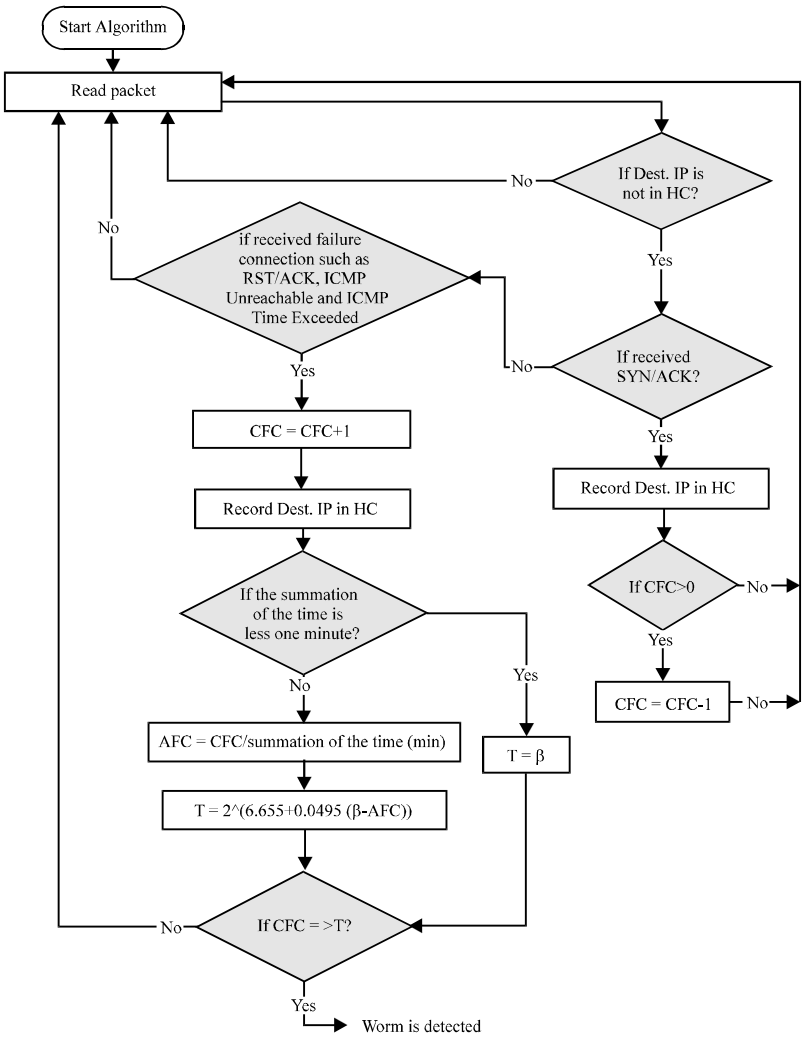

Fig. 10: The Flowchart Diagram for DPT

Table 2: Speed of detection between DPT and Yang *et al.* (2006)

Average of failure connection/min	Threshold in DPT	Detection time in DPT	Threshold in Yang's	Detection time in Yang's
100	104	1 min 2 sec	3001	30 min 1 sec
99	108	1 min 5 sec	3001	30 min 19 sec
98	112	1 min 9 sec	3001	30 min 37 sec
97	116	1 min 12 sec	3001	30 min 56 sec
96	120	1 min 15 sec	3001	31 min 16 sec
95	124	1 min 18 sec	3001	31 min 35 sec
94	128	1 min 22 sec	3001	31 min 56 sec
93	133	1 min 26 sec	3001	32 min 16 sec
92	137	1 min 29 sec	3001	32 min 37 sec
91	142	1 min 34 sec	3001	32 min 59 sec
90	147	1 min 38 sec	3001	33 min 21 sec
89	152	1 min 42 sec	3001	33 min 43 sec
88	157	1 min 47 sec	3001	34 min 6 sec
87	163	1 min 52 sec	3001	34 min 30 sec
86	169	1 min 58 sec	3001	34 min 54 sec
85	174	2 min 3 sec	3001	35 min 18 sec
84	181	2 min 9 sec	3001	35 min 44 sec
83	187	2 min 15 sec	3001	36 min 9 sec
82	193	2 min 21 sec	3001	36 min 36 sec
81	200	2 min 28 sec	3001	37 min 3 sec

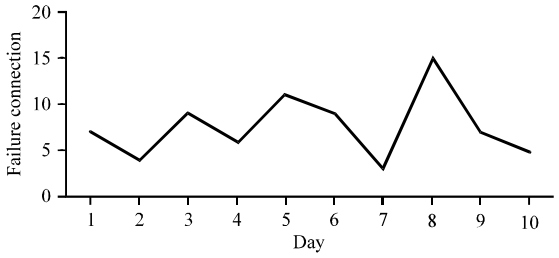

Fig. 11: DPT failure connection in normal computer

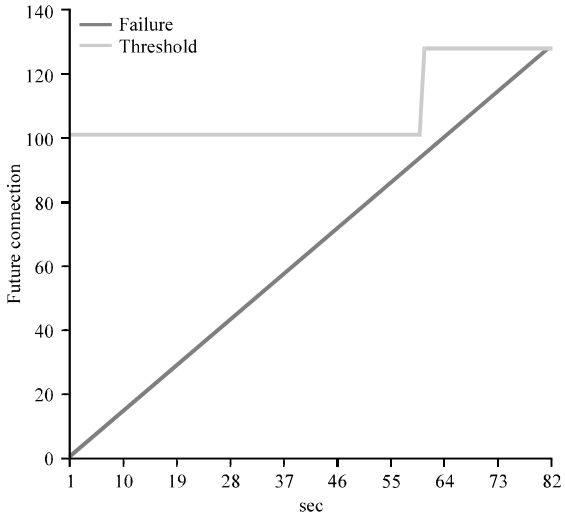

Fig. 12: DPT detected the worm after 82 sec

The result, maximum failure was 15 failure connections per day and the total for ten days was 76 failure connections. Moreover, DPT threshold for detecting the Internet was 101 failure connections per minute. DPT was examined for ten days and the result was not faced by any false-positive warning. The average of failure connection received was 7.6 failure connections per day by using DPT. It was a low failure because DPT considered only abnormal failure connection. The result of the experiment is shown in Fig. 11.

After measured the false-positive alarm, the study infected the machine by Ramen worm that used FIN/SYN flag to show the ability of DPT detection. Figure 12 and 13 show two experiments of Ramen worm detection. Figure 12 shows the average of failure connection which was 94/min and the time process to detect the worm, which was 82 sec. In Fig. 13, the average of failure connection was 90/min and the time process to detect the worm was 98 sec. DPT is more dynamic to detect the stealth worm because it calculates the threshold every minute as shown in Fig. 12 and 13 every minute has a new threshold.

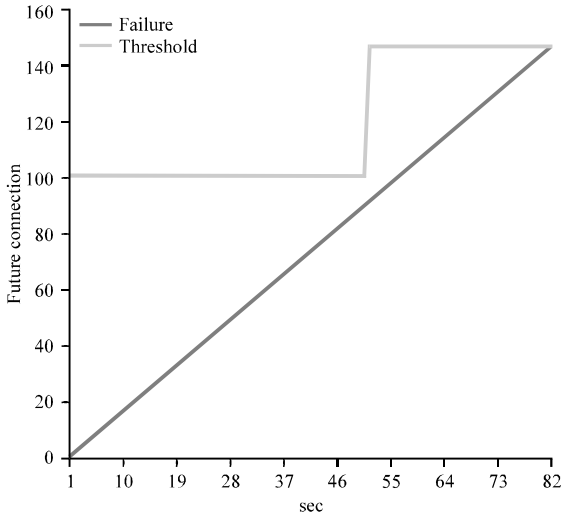

Fig. 13: DPT detected the worm after 98 sec

CONCLUSION

This study presented DPT for detecting the Internet scanning worms. The study focused only on TCP scanning worm. Furthermore, the worm is detected via depended on failure connections like RST/ACK, ICMP unreachable and ICMP Time Exceeded. The study found DPT faster than Yang *et al.* (2006). The future study will focus to detect the worm that used another protocol for scanning such as User Datagram Protocol (UDP).

REFERENCES

Berk, V., R. Gray and G. Bakos, 2003. Using sensor networks and data fusion for early detection of active worms. Proceedings of the SPIE AeroSense Conference, April 21-25, 2003, Orlando, Florida.

Chen, S. and Y. Tang, 2007. DAW: A distributed antiworm system. IEEE Trans. Parallel Distrib. Syst., 18: 893-906.

Costa, M., 2006. End-to-end containment of internet worm epidemics. Ph.D. Thesis, University of Cambridge.

De Vivo, M., E. Carrasco, G. Isern and G.O. de Vivo, 1999. A review of port scanning techniques. SIGCOMM Comput. Commun. Rev., 29: 41-48.

Dubendorfer, T., M. Bossardt and B. Plattner, 2005. Adaptive distributed traffic control service for DdoS attack mitigation. Proceedings of the 19th International Parallel and Distributed Processing Symposium, April 4-8, 2005, Washington, DC, USA.

Ellis, D.R., J.G. Aiken, K.S. Attwood and S.D. Tenaglia, 2004. A behavioral approach to worm detection. Proceedings of the 2nd ACM Workshop on Rapid Malcode, Washington DC, USA., October 29, 2004, ACM, New York, USA., pp: 43-53.

Fukushima, M. and S. Goto, 1999. Analysis of TCP flags in congested network. Proceedings of the 53rd Session of the International Statistical ESF Workshop, September 3, 1999, Dubrovnik, Croatia.

Hiestand, R., 2005. Scan detection based identification of worm-infected hosts. Technical Report, ETH, Zurich, April 2005.

Jiang, X. and X. Zhu, 2009. vEye: Behavioral footprinting for self-propagating worm detection and profiling. Knowledge Inform. Syst., 18: 231-262.

Lyon, G.F., 2009. Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning. Insecure Co., LLC, USA.

Meenakshi, S. and S.K. Srivatsa, 2007. A distributed framework with less false positive ratio against distributed denial of service attack. Inform. Technol. J., 6: 1139-1145.

Messer, J., 2007. Secrets of network cartography: A comprehensive guide to nmap. http://www.professormesser.com/

Min, F. and R. Gupta, 2009. Detecting virus mutations via dynamic matching. Proceedings of the IEEE International Conference on Software Maintenance, September 20-26, 2009, Edmonton, Canada, pp: 105-114.

Mohammed, M.M.Z.E., H.A. Chan, N. Ventura, M. Hashim, I. Amin and E. Bashier, 2010. Detection of zero-day polymorphic worms using principal component analysis. Proceedings of the 6th International Conference on Networking and Services, March 7-13, 2010, Cancun, Mexico, pp: 277-281.

Moskovitch, R., C. Feher and Y. Elovici, 2009. A Chronological Evaluation of Unknown Malcode Detection. In: Intelligence and Security Informatics, Chen, H., C. Yang, M. Chau and S.H. Li (Eds.). Vol. 5477, Springer-Verlag, New York, USA., pp: 112-117.

Muda, Z., W. Yassin, M.N. Sulaiman and N.I. Udzir, 2011. A K-means and naive bayes learning approach for better intrusion detection. Inform. Technol. J., 10: 648-655.

Nasir, M.H.N.M., N.H. Hassan and S.S.M. Fauzi, 2008. Protecting windows registry directory and hence increasing the security level of the windows operating system. Inform. Technol. J., 7: 840-849.

Paul, L.C., 2001. Code red: A field study of a worm in the wild. Global Information Assurance Certification Paper.

Tang, Y. and S. Chen, 2005. Defending against internet worms: A signature-based approach. Proceedings of the IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Volume 2, March 13-17, 2005, Miami, FL., USA., pp: 1384-1394.

Yang, X., J. Lu, Y. Zhu and P. Wang, 2006. Simulation and evaluation of a new algorithm of worm detection and containment. Proceedings of the 7th International Conference on Parallel and Distributed Computing, Applications and Technologies, December 4-7, 2006, Taipei, Taiwan, pp: 448-453.

Yaqub, K., 2006. Modeling security requirements of target of evaluation and vulnerabilities in UML. Master Thesis, Lulea Tekniska Universitet, Sweden.

Yu, H., M.X. He and H.C. Sun, 2009. The design of firewall in mobile phone based on cross-layer collaboration. Inform. Technol. J., 8: 1049-1053.

Zolkipli, M.F. and A. Jantan, 2010. A framework for malware detection using combination technique and signature generation. Proceedings of the 2nd International Conference on Computer Research and Development, May 7-10, 2010, Kuala Lumpur, Malaysia, pp: 196-199.

Zou, C.C., L. Gao, W. Gong and D. Towsley, 2003. Monitoring and early warning for internet worms. Proceedings of the 10th ACM Symposium on Computer and Communication Security, October 27-30, 2003, ACM, New York, USA., pp: 190-199.